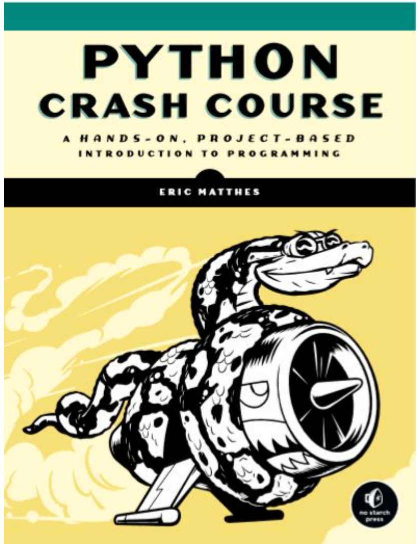
SALAR CODE

PYTHON

Lecturer

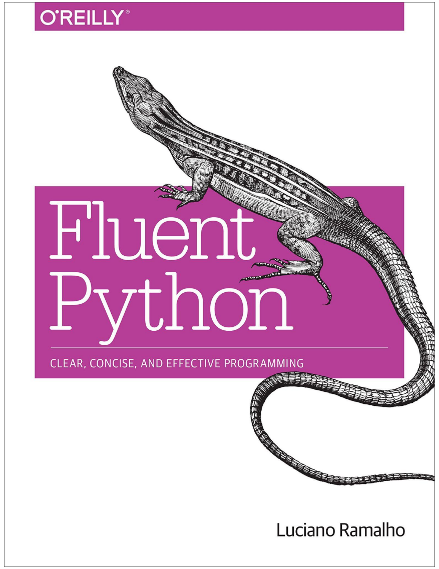
Salar Mokhtari Laleh

Reference

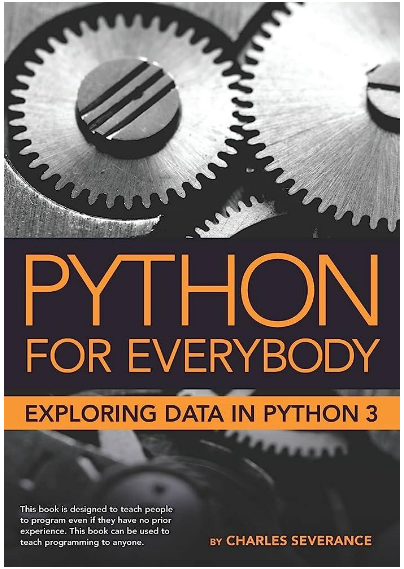


Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming

Author: Eric Matthes



Fluent Python
Author: Luciano Ramalho


PYTHON
FOR EVERYBODY
EXPLORING DATA IN PYTHON 3
This book is designed to teach people to program even if they have no prior experience. This book can be used to teach programming to anyone.
BY CHARLES SEVERANCE

Python for Everybody
Author: Charles Severance

History

Python was conceived in the late 1980s.by Guido van Rossum at Centrum Wiskunde & Informatica (CWI) in the Netherlands as a successor to the ABC programming language, which was inspired by SETL,capable of exception handling and interfacing with the Amoeba operating system.Its implementation began in December 1989.Van Rossum shouldered sole responsibility for the project, as the lead developer, until 12 July 2018, when he announced his "permanent vacation" from his responsibilities as Python's "benevolent dictator for life", a title the Python community bestowed upon him to reflect his long-term commitment as the project's chief decision-maker.In January 2019, active Python core developers elected a five-member Steering Council to lead the project.

Python 2.0 was released on 16 October 2000, with many major new features such as list comprehensions, cycle-detecting garbage collection, reference counting, and Unicode support.Python 3.0, released on 3 December 2008, with many of its major features backported to Python 2.6.x and 2.7.x. Releases of Python 3 include the ‘2to3’ utility, which automates the translation of Python 2 code to Python 3.

‘Guido van Rossum’

Why Python?

- Easy to Learn and Use
- Handy for Web Development Purposes
- The Language is Extensively used in Data Science
- Hundreds of Python Libraries and Frameworks
- Python can be used in ML(machine learning) tool
- Flexibility and Reliability
- Python Automates Tasks
- The First-choice Always
- ETC

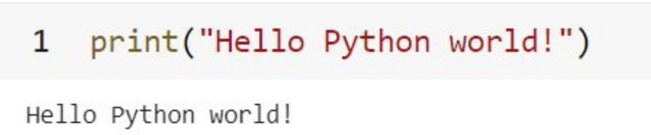
02

Variables and Simple Data Types

‘Hello_world.py’ → Let's take a closer look at what Python does when running a Python file.

- ‘py’ → a program file or script written in Python, an interpreted object-oriented programming language

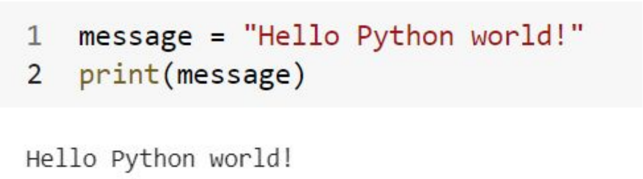
It can be created and edited with a text editor, but requires a Python interpreter to run. PY files are often used to program web servers and other administrative computer systems.



```python
print("Hello Python world!")
```

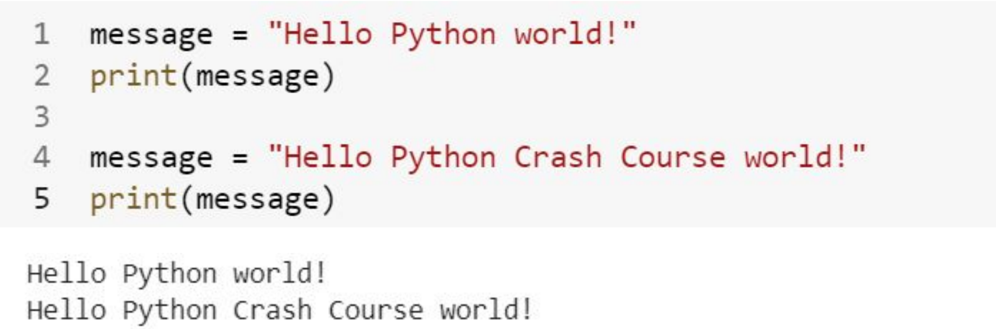
```
Hello Python world!
```

Variables



```python
message = "Hello Python world!"
print(message)
```

```
Hello Python world!
```



```python
message = "Hello Python world!"
print(message)

message = "Hello Python Crash Course world!"
print(message)
```

```
Hello Python world!
Hello Python Crash Course world!
```

Naming and Using Variables

- Variable names can contain only **letters**, **numbers**, and **underscores**.They can **start** with a letter or an underscore, but not with a **number**.For instance, you can call a variable ‘`message_1`’ but not ‘`1_message`’.

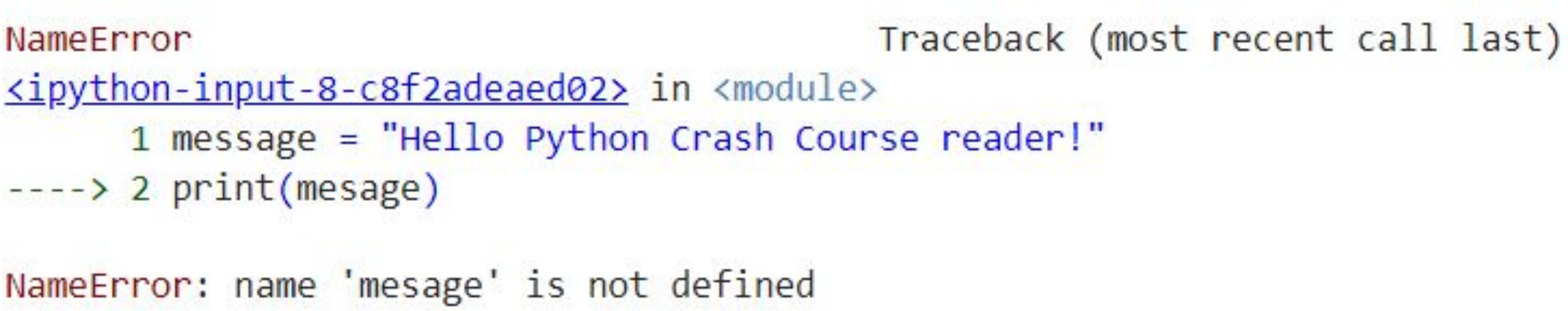
  ```
  SyntaxError: invalid decimal literal
  ```

- Spaces are **not allowed** in variable names, but underscores can be used to separate words in variable names. For example, ‘`greeting_message`’ works, but ‘`greeting message`’ will cause errors.

  ```
  SyntaxError: invalid syntax
  ```

- Avoid using Python keywords and function names as variable names; that is, do not use words that Python has reserved for a particular programmatic purpose, such as the word ‘`print`’.
- Variable names should be short but descriptive. For example, ‘`name`’ is better than ‘`n`’, ‘`student_name`’ is better than ‘`s_n`’, and ‘`name_length`’ is better than ‘`length_of_persons_name`’.
- Be careful when using the lowercase letter ‘`l`’ and the uppercase letter ‘`O`’ because they could be confused with the numbers ‘`1`′ and ‘`0`′ .

Avoiding Name Errors When Using Variables



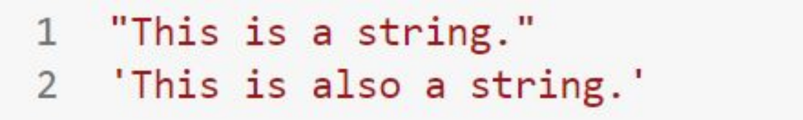
```
NameError                                 Traceback (most recent call last)
<ipython-input-8-c8f2adeaed02> in <module>
      1 message = "Hello Python Crash Course reader!"
----> 2 print(mesage)

NameError: name 'mesage' is not defined
```

String

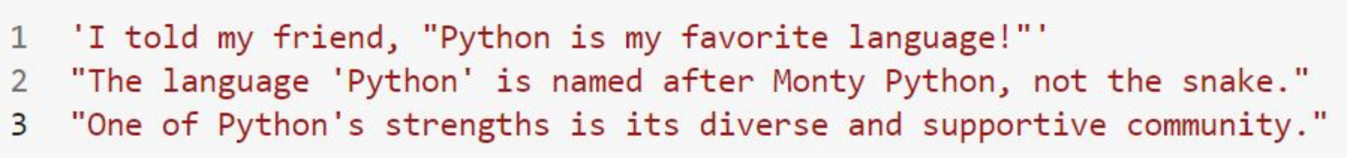
String

A ‘`string`’ is a series of characters. Anything inside quotes is considered a string in Python, and you can use single or double quotes around your strings



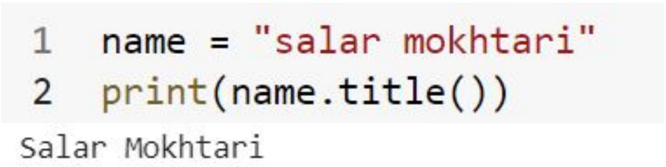
```
"This is a string."
'This is also a string.'
```

This flexibility allows you to use quotes and apostrophes within your strings



```
'I told my friend, "Python is my favorite language!"'
"The language 'Python' is named after Monty Python, not the snake."
"One of Python's strengths is its diverse and supportive community."
```

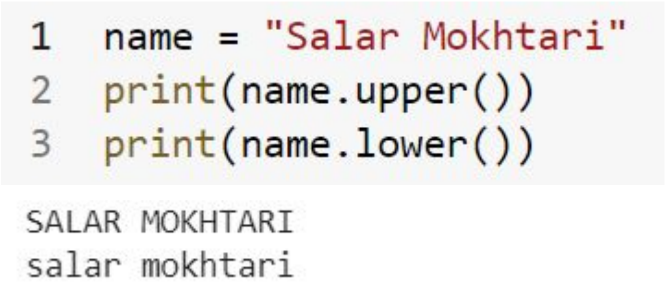
Changing Case in a String with Methods



```python
name = "salar mokhtari"
print(name.title())
```

```
Salar Mokhtari
```

➔ Title

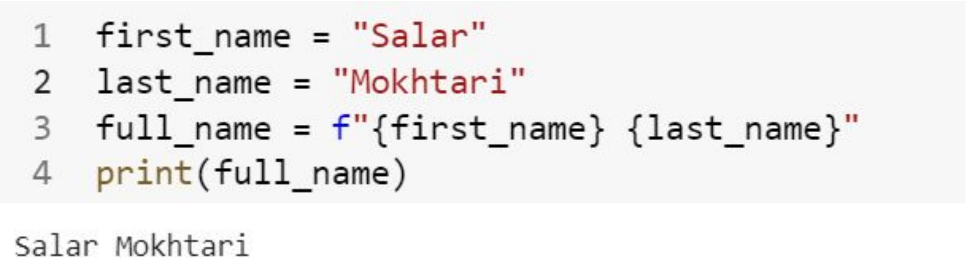


```python
name = "Salar Mokhtari"
print(name.upper())
print(name.lower())
```

```
SALAR MOKHTARI
salar mokhtari
```

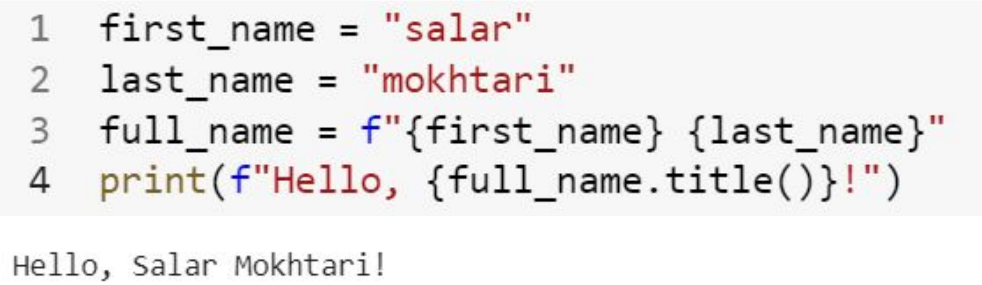
➔ Upper & Lower

Using Variables in Strings



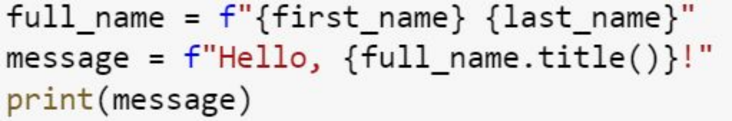
```
first_name = "Salar"
last_name = "Mokhtari"
full_name = f"{first_name} {last_name}"
print(full_name)
```

```
Salar Mokhtari
```



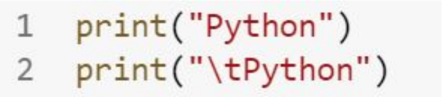
```
first_name = "salar"
last_name = "mokhtari"
full_name = f"{first_name} {last_name}"
print(f"Hello, {full_name.title()}!")
```

```
Hello, Salar Mokhtari!
```

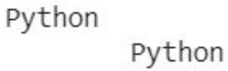


```
full_name = f"{first_name} {last_name}"
message = f"Hello, {full_name.title()}!"
print(message)
```

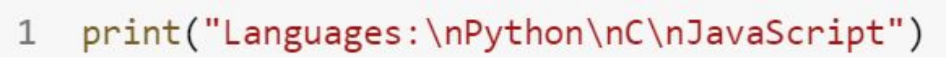
Adding Whitespace to Strings with Tabs or Newlines



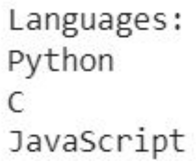
```python
print("Python")
print("\tPython")
```



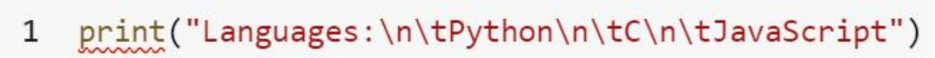
```
Python
        Python
```



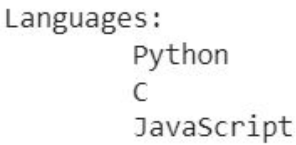
```python
print("Languages:\nPython\nC\nJavaScript")
```



```
Languages:
Python
C
JavaScript
```

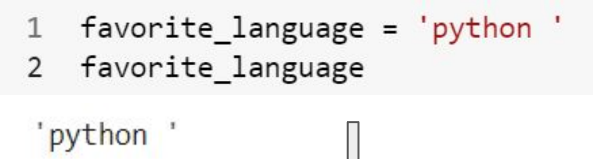


```python
print("Languages:\n\tPython\n\tC\n\tJavaScript")
```



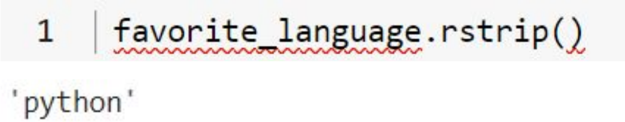
```
Languages:
        Python
        C
        JavaScript
```

Stripping Whitespace



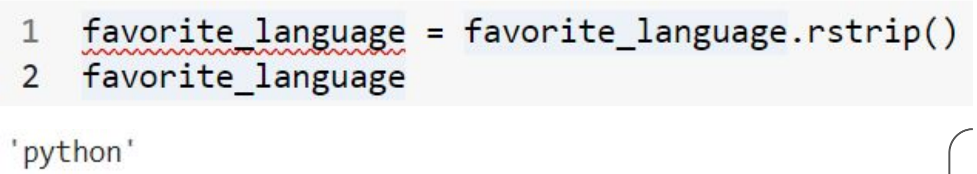
```
favorite_language = 'python '
favorite_language
```

```
'python '
```



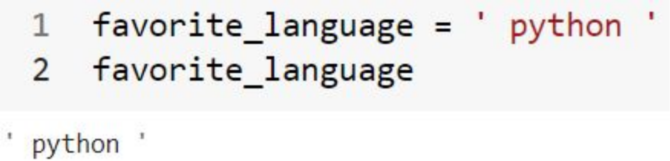
```
favorite_language.rstrip()
```

```
'python'
```



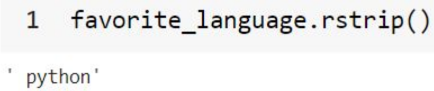
```
favorite_language = favorite_language.rstrip()
favorite_language
```

```
'python'
```



```
favorite_language = ' python '
favorite_language
```

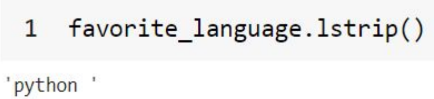
```
' python '
```



```
favorite_language.rstrip()
```

```
' python'
```

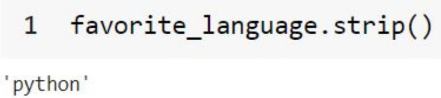
Right



```
favorite_language.lstrip()
```

```
'python '
```

Left



```
favorite_language.strip()
```

```
'python'
```

All

Q - 1.1) Is there an error in the following code?

```
Message_1 = "One of Python's strengths is its diverse community."
Message_2 = 'One of Python's strengths is its diverse community.'

print(Message_1)
print(Message_2)
```